

REMARKS

In response to the Restriction Requirement mailed 10/04/2002, Applicant respectfully requests that the following claim group be selected for prosecution of the above-identified application:

Group I, claims 1 - 6, drawn to a method of partitioning a program into blocks of code, and optimizing said program by reordering the blocks of code, classified in class 717, subclass 159.

Claims 1 - 6 are pending for examination in the referenced application. Claims 7 - 9 are hereby cancelled without prejudice.

CONCLUSION

Applicants respectfully submit the present application is in condition for allowance. If the Examiner believes a telephone conference would expedite or assist in the allowance of the present application, the Examiner is invited to call John Ward at (408) 720-8300, x237.

Authorization is hereby given to charge our Deposit Account No. 02-2666 for any charges that may be due.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN

A handwritten signature in black ink, appearing to read 'John P. Ward', is written over a horizontal line.

John P. Ward
Reg. No. 40,216

Date: 01/03/2003

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, CA 90025-1026
(408) 720-8300

ATTACHMENT A

Claims 1 - 6 are pending, and Claims 7 - 9 are hereby cancelled without prejudice.

A marked-up version of the amended claims is as follows:

1. For a computer-executable program that operates on a data structure, where the data structure must have a required state at selected program points, a method of transforming said program comprising the steps of:

(A) analyzing the program to determine the state of said data structure at said selected program points;

(B) partitioning said determined state at each said program point into components that may each be set separately;

(C) determining the operations required to set each component of the state at each selected program point; and

(D) placing said operations in a way that eliminates partial redundancies of said operations.

4. The method of claim 1, wherein the data structure stores items on a first-in last-out basis.

5. The method of claim 2, wherein the states of the data structure are represented as paths on a tree of nodes where:

(A) each path traverses the tree towards the root, and

(B) each node on the path represents a component of the state.

4. The method of claim 2, wherein the data structure represents actions to be taken by the program if an exceptional situation arises.

5. The method of claim 4, wherein the selected program points are the points of execution immediately before instructions that might cause an exceptional situation.

6. The method of claim 5, wherein the actions to be taken are represented explicitly as exceptional paths in a graph before the transformation, and said exceptional paths are removed.

7. (cancelled) ✓

8. (cancelled) ✓

9. (cancelled) ✓